Clustering with a new distance measure based on a dual-rooted tree

Laurent Galluccio, Olivier J.J. Michel, Pierre Comon, Mark Kliger, Alfred O. Hero

Clustering with a new distance measure based on a dual-rooted tree

Laurent Galluccio[a], Olivier Michel[b], Pierre Comon[b,*], Mark Kliger[c], Alfred O. Hero[d]

[a] *Lab. Lagrange, Observatoire de la Cote d'Azur, BP.4229 06304 NICE Cedex 4, France*
[b] *Gipsa-Lab UMR 5216, 961 rue de la Houille Blanche - BP.46 - 38402 Saint Martin d'Hères Cedex, France*
[c] *Medasense Biometrics Ltd., Ofakim, Israel*
[d] *Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Avenue, Ann Arbor MI 48109-2122, USA*

Abstract

This paper introduces a novel distance measure for clustering high dimensional data based on the hitting time of two Minimal Spanning Trees (MST) grown sequentially from a pair of points by Prim's algorithm. When the proposed measure is used in conjunction with spectral clustering, we obtain a powerful clustering algorithm that is able to separate neighboring non-convex shaped clusters and to account for local as well as global geometric features of the data set. Remarkably, the new distance measure is a true metric even if the Prim algorithm uses a non-metric dissimilarity measure to compute the edges of the MST. This metric property brings added flexibility to the proposed method. In particular, the method is applied to clustering non Euclidean quantities, such as probability distributions or spectra, using the Kullback-Liebler divergence as a base measure. We reduce computational complexity by applying consensus clustering to a small ensemble of dual rooted MSTs. We show that the resultant consensus spectral clustering with dual rooted MST is competitive with other

*Corresponding author
Email addresses: laurent.galluccio@gmail.com (Laurent Galluccio), olivier.michel@gipsa-lab.inpg.fr (Olivier Michel), pierre.comon@grenoble-inp.fr (Pierre Comon), mark@medasense.com (Mark Kliger), hero@umich.edu (Alfred O. Hero)

clustering methods, both in terms of clustering performance and computational complexity. We illustrate the proposed clustering algorithm on public domain benchmark data for which the ground truth is known, on one hand, and on real-world astrophysical data on the other hand.
Key words: Non-metric clustering; minimal spanning tree; Prim's algorithm; affinity measure; co-association measure; consensus clustering

---

1. Introduction

The process of clustering partitions a set of data into non-overlapping subsets. The partitions are determined such that patterns belonging to the same cluster share more similarity with each other than with patterns belonging to different clusters [30]. Such problems have been investigated in many fields of research including: data mining [6], pattern recognition [46], image segmentation [50], computer vision [41] and bio-informatics [52]. There are a wide range of clustering methods available, *e.g.* , hierarchical clustering, spectral clustering, graph partitioning algorithms and k-means [25, 26, 31, 46]. In this paper we introduce a new clustering method that uses dual rooted trees combined with consensus methods. The approach is closely related to level-set methods [44] and entropy minimization [27]. However, dual rooted trees have advantageous mathematical properties and their performance is competitive with the state-of-the art.

Dissimilarity measures between data points play a crucial role in designing clustering algorithms. These measures determine how the clustering algorithm differentiates pairs of points within the same cluster (high similarity) from pairs of points in different clusters (low similarity). In many cases using Euclidean metric to measure dissimilarities between data points is insufficient. This has motivated spectral diffusion methods of clustering [34, 37, 42, 51]. The original spectral method used a Gaussian kernel on a Euclidean metric to construct a

more discriminating dissimilarity measure [37]. In [34], the Gaussian kernel was interpreted as a heat diffusion kernel which induces a random walk on the graph with nodes consisting of data points, yielding a measure of dissimilarity. In [39] a "commute time" dissimilarity measure is introduced, and is closely related to diffusion distance. In both of these approaches the diffusion and commute time dissimilarity measures are used for embedding data points into a new system of coordinates defined by the eigenvectors of the heat kernelized affinity matrix. The final clustering step is achieved by using k-means on the embedded data.

The dual rooted minimal spanning tree (MST) clustering approach proposed in this paper is different. Starting from a base dissimilarity measure between data points, it constructs MSTs rooted at different points in the dataset. It then defines the dissimilarity between pairs of points as the time it takes before collision of the two MSTs as they are grown from each root using Prim's algorithm [38]. This time is called the dual rooted tree hitting time, and it is a non-Euclidean dissimilarity measure that describes global as well as local geometrical properties of the data set, as explained in details in Section 2.2. In particular, this *hitting time* can be used as a measure of dissimilarity between the two roots and it is influenced by the distance between the roots in addition to the dissimilarity of their local neighborhoods. The matrix of pairwise dissimilarities can then be transformed into an affinity matrix by applying the standard heat kernel approach used in spectral clustering. This principal role of the local neighborhoods of each pair of points is one of the main differences between the dual rooted MST approach and the diffusion kernel and commute time methods.

The starting point for this paper is the simple algorithm described above, called the Symmetric Dual Rooted Prim Tree (SDRPT) algorithm, introduced by two of the authors of this paper [24]. It computes the hitting time for all

$\binom{N}{2}$ pairs of points, the two rooted MSTs are grown in parallel simultaneously from each root, and it results in a pair of rooted MSTs that have the same number of edges at the hitting time. Building on the SDRPT concept we then define a modified algorithm, called the Dual Rooted Prim Tree (DRPT), that results in a pair of MSTs having different numbers of edges at the hitting time. Specifically, it selects a randomly chosen subset of pair of roots and grows MSTs sequentially and asymmetrically: at each stage of the Prim's algorithm, among the two new edges proposed for each MST only the rooted MST with the smallest edge is grown. Moreover, instead of using the "hitting time" as a dissimilarity measure, the length of the last constructed edge is used. This latter edge is a clique separator: its removal from the final graph disconnects the two rooted trees. The DRPT and the SDRPT have substantially different properties. In particular, the dissimilarity measure produced by the DRPT is a true metric regardless of the base dissimilarity measure used to define edge lengths for the Prim MST constructions.

Since the computation of the DRPT for all $\binom{N}{2}$ pair of vertices is necessary to construct a complete dissimilarity matrix, it may have a prohibitive computational cost. To address this point, we propose to create a consensus affinity matrix [33] based on the clusters produced by a subset of $M \ll \binom{N}{2}$ DRPT rooted at random pairs of points. As in the SDRPT of [24], or for consensus matrices in [53] spectral clustering to this matrix can then be used. Consensus clustering is a method for merging results from different algorithms, or from different clustering realizations associated with different initial conditions. This concept finds its origin in multi-classifier and multi-learner systems (see [23] and [33] for a brief history). The main idea is to empirically estimate performance by data partitioning, to create a set of clustering realizations that can be compared and combined [9, 16, 43]. The methods of cross-validation, bagging and

boosting classifiers [10, 18] are examples. We apply consensus clustering to dual rooted MSTs by applying it to the random selection of pairs of roots. As the proposed method accumulates evidence for clustering from each of the DRPTs, we refer to it as "Evidence Accumulating Clustering with Dual rooted Prim tree Cuts" (EAC-DC).

The EAC-DC approach has several features that we summarize here. First, the DRPT dissimilarity measure captures the dissimilarity of the MST neighborhoods of each pair of points. Second, since only a smaller random subset of pairs are used in the DRPT, it benefits from lower computational complexity than SDRPT with spectral clustering. We show by simulation and experiment that the EAC-DC outperforms state-of-the-art clustering methods on benchmark data sets. Third, as proven in the sequel, regardless of the base dissimilarity measure adopted to build the MST, the DRPT produces a dissimilarity measure which is a metric and this property can translate into improved performance relative to the SDRPT with spectral clustering. To illustrate this property, the DRPT is implemented on the symmetrized KL divergences between pairs of infrared star spectra to cluster stars in an astrophysical dataset.

The outline of the paper is as follows. Section 2 provides a brief introduction to minimal spanning trees and Prim's algorithm for general dissimilarity measures. In Section 2.2 dual rooted MST are discussed, the DRPT is proposed and its properties are discussed. Consensus clustering is applied to the DRPT dissimilarity measure in Section 3. Implementation and computational issues are also discussed in this section. Finally, after a brief review of clustering performance measures, an extensive comparative study is presented for both simulated and real datasets from the UCI repository of machine learning [2] and an astrophysical dataset for star classification.

2. Minimal Spanning Trees and Prim's algorithm

Let $V = \{v_1, v_2, \ldots, v_N\}$ denote a set of data points in $\mathbb{R}^l$, representing feature vectors.

2.1. Construction of MST

Define $G = (V, E)$ the undirected graph where $E = (e_{ij} : e(v_i, v_j), (i, j) \in (1, \ldots, N))$ denotes a set of undirected edges between vertices (data points) V. Given a base dissimilarity measure $w(v, u)$ between data points v, u the weight of an edge is defined as $w_{ij} = W(e_{ij}) = w(v_i, v_j)$. The weight w_{ij} measures the dissimilarity or separation between two vertices. It will be assumed that the base dissimilarity measure is symmetric, positive and homogeneous, i.e., $w(v_i, v_j) = w(v_j, v_i)$ and $w(v_i, v_i) = 0$, but it does not have to be a metric. A common choice for the base dissimilarity is the Euclidean length. Although it enjoys many attractive features [27], more general base dissimilarity measures are considered in this paper.

A spanning tree $\mathcal{T}$ through the set of vertices V is a connected acyclic graph that passes through all the N vertices v_i, $i \in \{1, \ldots, N\}$ in the set. The weight of the tree $\mathcal{T}$ is the sum of all edge weights. The minimal spanning tree (MST) is the tree which has the minimal weight

$$\mathcal{W}_N(V) = \min_{\mathcal{T}} \sum_{e_{ij} \in \mathcal{T}} w_{ij}$$

We apply Prim's algorithm [38] to construct the MST.

2.2. Dual Rooted Prim Tree (DRPT)

In the Symmetric Dual Rooted Prim Tree (SDRPT) method proposed in [24], Prim's algorithm is used to construct a pair of rooted MST that are separately and simultaneously grown. A graph-based distance measure between

two vertices can be derived from the hitting time (number of steps taken until trees collide) of the two rooted MSTs rooted at each pair of distinct vertices. It is important to emphasize that both trees are grown in parallel at each time step. An important drawback of SDRPT is that it builds candidate clusters that always have the same number of vertices. Hence, in realistic cases where clusters have different sizes, SDRPT introduces a bias, which leads to degraded performances.

In contrast, in the DRPT construction, each rooted MST competes for growth. At each time step the pair of Prim algorithms produces a pair of candidate edges, one for each rooted MST, and only the MST with minimum length candidate edge is grown. The algorithm is again stopped when the two rooted MSTs collide; with this modification, the two MSTs may not have an equal number of edges. An example is presented in Figure 1.

For a given pair of vertices $\{v_1, v_2\}$ serving as roots of two rooted MSTs $\mathcal{T}_1$ and $\mathcal{T}_2$, let w_{final} be the weight of that final connected edge. This final edge connects $\mathcal{T}_1$ to $\mathcal{T}_2$. We define a new distance measure between v_1 and v_2 as

$$\delta(v_1, v_2) = w_{final}$$

It is important to stress that this measure depends upon the MST topology, and should not be confused with the dissimilarity $w(.,.)$ used to grow the rooted MSTs. The tree obtained by the union of the two rooted MSTs is referred to as **D**ual **R**ooted **P**rim **T**ree (DRPT), to emphasize its similarities with the Prim construction method, which connects a single vertex at each iteration.

The DRPT (Fig. 1) has interesting properties, subsequently used:

P1: For a given pair of root vertices $\{v_1, v_2\}$, the last constructed edge, which connects the two rooted MSTs together, is always the largest (with maximum edge weight w_{final}) among all edges in both rooted MSTs (see

Appendix A1).

P2: Recall that $\delta(v_1, v_2) = w_{final}$ for the DRPT rooted at v_1 and v_2. Then, $\delta(v_1, v_2)$ is a metric, even if the base dissimilarity measure $w(v, u)$ is not (see Appendix A1 for a proof).

P3: When the weights $\{w_{ij}\}_{i>j}$ are unique, the DRPT rooted at v_1 and v_2 is the MST for the subset of vertices spanned by the DRPT (the MST is unique and does not depend upon the root used to initialize Prim's algorithm).

P4: Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be the MSTs rooted at v_1 and v_2, stopped at the hitting time. The DRPT satisfies (see Appendix A2).

P4a: The DRPT metric $\delta(x, y)$ is constant over $x \in \mathcal{T}_1$ and $y \in \mathcal{T}_2$:

$$\forall (x, y) \in \mathcal{T}_1 \times \mathcal{T}_2, \qquad \delta(x, y) = \delta(v_1, v_2)$$

P4b: The DRPT metric between any two vertices from $\mathcal{T}_1$ (resp. $\mathcal{T}_2$) is upper bounded by $\delta(v_1, v_2)$:

$$\forall (x, y) \in [\mathcal{T}_1 \times \mathcal{T}_1] \cup [\mathcal{T}_2 \times \mathcal{T}_2], \qquad \delta(x, y) \leq \delta(v_1, v_2)$$

P5: Let $\mathcal{R}_{v_2}^{v_1}$ stand for the relation, defined relatively to v_1 and v_2 for any $x, y \in V$ by

$$x \mathcal{R}_{v_2}^{v_1} y \text{ if } \delta(x, y) \leq \delta(v_1, v_2).$$

Then $\mathcal{R}_{v_2}^{v_1}$ is trivially symmetric and reflexive. Transitivity of $\mathcal{R}_{v_2}^{v_1}$ is easily obtained as a consequence of properties $P2$ and $P4$ (see Appendix A3). Therefore $\mathcal{R}_{v_2}^{v_1}$ is an equivalence relation.

Discussion From the preceding properties, we easily infer the following: As only the dissimilarities between vertices from either $\mathcal{T}_1$ or $\mathcal{T}_2$ are considered, and not the convexity of $\mathcal{T}_1$ or $\mathcal{T}_2$, all properties P1 to P5 hold for either convex or non convex subsets $\mathcal{T}_1$ and $\mathcal{T}_2$.

As $\mathcal{T}_1$ (resp. $\mathcal{T}_2$) rooted at v_1 (resp. v_2) are grown until the largest possible separation between $\mathcal{T}_1$ and $\mathcal{T}_2$ is found, $\mathcal{T}_1$ (resp. $\mathcal{T}_2$) includes all vertices that may be connected to v_1 (resp v_2) by a path that contains edges all shorter than $w_{final} = \delta(v_1, v_2)$.

A consequence of *P4a* and *P4b* is that the distance $\delta(x, y)$ accounts for global geometrical properties. Moreover, since it is constructed by growing subtrees, it also takes into account local geometry.

Since $\delta(.,.)$ is a metric, data may easily be embedded in a system of Euclidean coordinates obtained by metric MultiDimensional Scaling (MDS) [7, 48]. That $\delta(.,.)$ satisfies the triangular inequality is not requested to compute a MDS embedding of the data, however this leads to embeddings that are nicer and easier to interpret [12].

Figures 2 and 3 show the results of the DRPT approach for a set of 10 different clustering problems. The data sets are chosen to reproduce the same difficulties as the benchmark data sets used in [55]. For each data set, the left side subgraph represents the vertices in the Euclidean plane. The middle subgraph shows the MDS based vertices representation constructed from the dissimilarity matrix $[D]_{ij} = \delta(v_i, v_j)$. In these toy-examples, the dissimilarity measure w used to grow the MSTs was the Euclidean metric. The right subgraph describes the clusters obtained by application of a classical k-means approach on the MDS embedded data shown in the middle graph. It is important to emphasize that k-means may be used here for clustering as it processes a data set (obtained by MDS) embedded in an Euclidean space. These embedded data

exhibit highly concentrated clusters: By property P4a and P5, both the intra and inter clusters measures will vary over a very restricted set of values. Most importantly, all n_1 vertices from a subtree $\mathcal{T}_1$ are separated from any vertex from a subtree $\mathcal{T}_2$ by the same distance. This leads to highly concentrated well separated clusters in the representation space associated with this new measure, as *e.g.* it appears on most tested data sets. This behavior, as well as properties P1 to P5, is not affected by possible non convexity of the clusters.

The results illustrate the ability of the proposed approach to handle segmentation of non convex clusters, and its behavior with respect to the presence of outliers. If edges that may connect (in a MST) outliers to vertices of a clusters remain larger than or equal to the length of 'in-cluster' edges, the clustering remains effective as illustrated on data set (A). If on the contrary the outliers form 'bridges' of edges with short length between clusters, the algorithm may fail to detect the clusters as in data set (J). This sensitivity to outliers as well as the computational burden, namely $\binom{N}{2}$ dual rooted tree instances to built the DRPT distance matrix, constitute the basic motivation for deriving an alternate solution in the next section.

2.3. From DRPT towards consensus clustering

For each pair of roots v_1 and v_2, the DRPT forms a MST over the set of vertices spanned by the DRPT. Property $P1$ above states that the largest edge is also the final connected one. The sets of vertices involved in $\mathcal{T}_1$ and $\mathcal{T}_2$ define candidate clusters containing v_1 and v_2, respectively. As the DRPT may not span the entire set of points V, we define a rejection cluster as the set of non connected vertices. Property $P5$ above states that these clusters are equivalence classes for the relation $\mathcal{R}_{v_2}^{v_1}$.

Let the DPRT be applied M times by random drawing of M root pairs from V. Let $P_i = \{C_{1i}, C_{2i}\}$ be the resulting partition of the set of connected

vertices for one of these root pairs $\{v_{1i}, v_{2i}\}$ (i indexes this particular choice of roots). Then $C_{1i} \cup C_{2i} \subseteq V$ is the neighborhood of vertices v_{1i} and v_{2i} and $V \setminus \{C_{1i} \cup C_{2i}\}$ is the rejection cluster. We define the DRPT pre-clustering algorithm as follows:

- Choose a set of M randomly chosen pairs of vertices.
- Compute the Dual rooted MST for each pair and construct two clusters by cutting the last (also the largest) edge.

The procedure is summarized in Algorithm 1. This pre-clustering algorithm will produce three clusters, denoted by $P_k = \{C_1^k, C_2^k, C_3^k\}$, for each root pair, $k = 1, \ldots, M$, giving a cluster ensemble $\{P_k\}_{k=1}^M$. In the next section we apply consensus clustering to this cluster-ensemble to form the proposed DRPT-based clustering algorithm.

Algorithm 1 DRPT pre-clustering algorithm

Input: V be a set of N data points $\in \mathbb{R}^l$
Ouput: $\mathbf{P} = (P_1, \ldots, P_M)$

1: **for** $k = 1$ to M **do**
2: Choose randomly two vertices v_i and v_j among V. For both vertices, compute a rooted greedy MST. Let $\mathcal{T}_i$ and $\mathcal{T}_j$ be the sets of points in the two rooted MSTs.
Initialization step: $\mathcal{T}_i = \{v_i\}$, $\mathcal{T}_j = \{v_j\}$.
3: **repeat**
4: Find closest non-connected vertex for both rooted MSTs.
$z_i = \underset{z \in V \setminus T_i}{argmin}\ w(z, T_i)$, $z_j = \underset{z \in V \setminus T_j}{argmin}\ w(z, T_j)$
5: Add the point z to its respective tree which has the shortest dissimilarity measure among the two candidates.
6: **until** the two rooted MSTs collide.
7: Cut the resulting tree at the largest edge.
8: Form three clusters, C_1^k, C_2^k corresponding to the two subsets of points identified by the cut and C_3^k contaning the remaining points (rejection cluster). This yields $P_k = \{C_1^k, C_2^k, C_3^k\}$.
9: **end for**

Computational cost of implementation

The computation of Algorithm 1 above is dominated by the Prim algorithm for computing DRPT distances. If all pairs of vertices were considered as roots for the DRPT tree, a brute force computation would run the Prim MST construction $N(N-1)/2$ times. This may become rapidly prohibitive for large N. However, computational complexity can be reduced by exploiting the fact that the full MST only needs to be run once: any two points connected in any MST subtree are also connected points in the MST. Furthermore, once the base dissimilarity matrix has been computed and stored, only the set of adjacency relations between vertices need to be recorded, and only logical operations are required to find the DRPT distance between any two points. Therefore, the full MST need only be computed once, and its descriptors can be stored in an array of size $2N$ (describing $N-1$ connections and $N-1$ edge lengths). Thus the DRPT algorithm cost remains of the same order as the cost of a full MST computation, and does not require any floating point operations. Yet, the Prim algorithm for constructing the full MST requires order $O(N \log N)$ operations.

As only the distances between neighboring data points are actually used in the Prim MST construction, we implemented a "Nearest Neighbor MST" [20] that easily scales to large data sets. This reduces the computational complexity of the distance matrix computation to $O(Nk \log k)$. Here k is typically much smaller than N, and never exceeds a bound on the maximum vertex degree of the MST. Finally, as only a few pairs of randomly chosen root vertices are needed to construct the co-association affinity matrix, only a small number of logical operations is required.

3. Evidence accumulating clustering with dual rooted Prim tree cuts (EAC-DC)

The computation and clustering performance of the above described DRPT pre-clustering algorithm can be improved by using consensus clustering. Consensus clustering was introduced to boost the performance of any arbitrary clustering algorithm. We briefly review the general method before specializing to the DRPT.

The goal of any clustering algorithm is to partition V into K clusters. Let $P_i = \{C_1, \ldots, C_K\}$ stand for a set of clusters obtained from the data by applying a clustering algorithm denoted $Algo_i$. Notice that $Algo_i$ and $Algo_j$ may be identical algorithms with different initialization parameters, or different clustering algorithms. In the proposed method, the clustering algorithm will be the same with different initialization parameters. M different partitions of the data will result and these are denoted $\mathbf{P} = \{P_1, \ldots, P_M\}$. In the context of this paper, these partitions are the DPRT partitions $P_k = \{C_1^k, C_2^k, C_3^k\}$ after its k-th run, as described in the previous section.

Each of the partitions in this cluster ensemble can be viewed as a "weak learner" of the true clusters in the data. These weak learners may individually have high sensitivity to noise and outliers, and perform poorly with proximal or interdigitated clusters. While pruning algorithms can be applied [1, 46, 54] we can do better and "amalgamate" the cluster ensemble to produce an improved clustering result. There have been many approaches to combine cluster ensembles including: multi-stage K-means [9], bagging [15], partitioning around medoids [32], quadrature mutual information consensus [47], graph representations [43], and cumulative voting [3]. Here we apply the evidence accumulation technique of Fred and Jain [16] [17], because of its well-known efficiency and simplicity to implement.

Specifically, the amalgamation of the cluster ensemble $\mathbf{P} = \{P_1, \ldots, P_M\}$ is

performed by the consensus clustering method [17], where we identify the co-association measure as the proportional number of runs of DPRT that classified a pair of points v_i, v_j in the same *non-rejection* cluster.

$$co\_assoc(v_i, v_j) = \frac{n(v_i, v_j)}{M}, \tag{1}$$

where $n(v_i, v_j)$ denotes the number of times both vertices were found in the same cluster (excluding the rejection cluster, C_3^k), over the M differently initialized DPRT runs. This definition of co-association is similar to others used in the literature [15, 16, 43] but other definitions are also possible, in particular, co-association that accounts for the rejection cluster (see our technical report [21] for details).

Once the co-association measure $co\_assoc(v_i, v_j)$ is defined for all v_i and v_j any clustering algorithm can be applied to determine the final clusters. In [16] hierarchical clustering algorithms are applied whereas in [43] a graph partitioning method is proposed. In [53], spectral clustering is applied on the consensus matrix.

The same idea is pursued here: in the present paper, a heat kernelized version of the affinity constructed above is used within a spectral clustering algorithm. The introduction of an exponential heat kernel of parameter σ (see equation 2) confers to the algorithm an improved robustness. This property was already quoted in [4] and was observed on all the experiments in this paper. The clusters will thus be identified by using the spectral clustering algorithm of Ng et al. [37], which extracts the eigen-structure of the affinity matrix A derived from the dissimilarity measure $\tau(v_i, v_j) = 1 - co\_assoc(v_i, v_j)$, and $\tau(v_i, v_i) = 0$. The ij-th element of A is

$$A(i,j) = \exp\left(-\frac{\tau(v_i, v_j)}{\sigma}\right) \propto \exp\left(\frac{co\_assoc(v_i, v_j)}{\sigma}\right), \forall (i,j) \in [1, N]^2 \tag{2}$$

where σ is a constant to be adjusted. The resulting clustering algorithm is called "Evidence Accumulating Clustering with Dual rooted prim tree Cuts" (EAC-DC). The basic steps of spectral clustering are recalled in Appendix A4.

Figures 4 and 5 provide illustrations of the application of the EAC-DC algorithm to the simulated data set studied previously. All clusters are correctly identified, as shown on subgraphs (e) and (f) for $M = \binom{N}{2}$ and $M = N/4$ respectively (see discussion about setting M, below). This again demonstrates the ability of EAC-DC to cluster non convex shapes, contrary to the results obtained with Euclidean distance: see rows I-J column (d). Moreover the clustering algorithm successfully discriminated against the uniform background of outliers, and hence provides an alternative to the method of [14] for dealing with outliers. This robustness to outliers can be attributed to the tendency of the outliers to belong to the rejection class during the consensus clustering stage of the EAC-DC algorithm.

Setting M: Choosing the lowest possible M leads to lower the computational load. However, a too low value of M may lead to poor performances. Any partition obtained from a dual rooted approach exhibit 3 clusters, of which 2 are rooted, and the third one contains those points that are not connected. If more clusters are presents, M should be chosen to insure that one point of each cluster will be selected as a root at least once. In the luckiest trivial case, where e.g. 2 well separated clusters are presents, and the pair of roots is such that there is a root in each cluster, even $M = 1$ is enough. On the contrary, if all M pairs considered are all members of the same clusters, the algorithm will fail to detect the clusters. The roots are chosen at random, and no general rule was exhibited so far, as the minimal M giving an acceptable clustering result heavily depends on the topology of the vertices (outliers, numbers of clusters, relative densities). On all presented experiments, partitions were added to the partition

set until all vertices appear connected at least once in the partition set, leading to $M < N/4 \ll \binom{N}{2}$ for all tested data sets. Therefore, most examples presented hereafter will use $N/4$ randomly set pairs of roots. Using more than $N/4$ initial partitions does not lead to any improvement, at least for the tested data.

Another issue that remains largely open is the proper choice of the parameter σ in the affinity matrix (2). As discussed in Luxburg [35], many rules of thumb have been proposed for selection of σ in spectral clustering but no firm theoretical justification of any of these heuristics yet exists. Most successful rules of thumb select σ proportional to the characteristic width of a cluster, where width is measured in the affinity domain. Most heuristics for setting σ are based on matching the characteristic spread of the dissimilarity kernel to the average width of of the clusters. For example, for the Gaussian kernel von Luxburg suggests setting σ to a fraction of the mean distance of a point to its k-th nearest neighbor, where e.g. $k = \log(N)+1$, which approximates the kernel width.

Such a rule for setting k is quoted by von Luxburg to be very ad hoc and its performance is highly dependent on the inter point distances of the data at hand. We used a simpler but similar heuristic in this paper: σ^2 in (2) is set to 1/10-th of the standard deviation of the measure : $\sigma^2_{meas} = 10^{-1}(N(N-1)-1)^{-1}\sum_{i>j}(\tau_{ij}-\overline{\tau})^2$, where $\overline{\tau} = (N(N-1))^{-1}\sum_{i>j}\tau_{ij}$. This heuristic is motivated by the fact that our new measure leads to highly concentrated and well separated clusters in the associated representation space: the intra variance of a cluster is much smaller than the inter-point distance variance. An illustration on real data is provided in Section 4.3. The sensitivity of the results with respect to σ was evaluated on BCW and Wine data sets (for which we know the true clustering) and for different performance indices, all described in the next section. Figure 6(a) shows that the choice of σ does not affect the

quality of the obtained clustering for BCW data and EAC_DC method. A stronger influence of σ is observed for the Wine data set (figure Figure 6(b)) and $EAC\_DC(KL)$ method, the best results are recorded for $.1 < \frac{\sigma}{\sigma_{meas}} < .2$.

Note that subgraphs (d) in figures 4 and 5 show the clustering results obtained by applying spectral clustering algorithm (cf. Appendix A4) on the complete Euclidean distance matrix. Many values of σ over the range $[0.05 \times \sigma_{meas}, 10 \times \sigma_{meas}]$ where tested and the best clustering was retained. For DRPT based approach, σ was set to $\sigma = .1 \times \sigma_{meas}$ for all data sets.

4. Tests and application

4.1. Non-metric base dissimilarity measures

To implement the DRPT pre-clustering algorithm, a base dissimilarity measure $w(v_i, v_i)$ is required. This base measure only needs to be symmetric and homogeneous; by Property P2 the DRPT algorithm will transform the base measure to a true metric. This gives us *considerable* flexibility for clustering various types of data, for which the 'natural' dissimilarity measure may not be a metric (see *e.g.* [8] for example of dissimilarity measures derived for categorical data). We present here some examples for which the 'natural' dissimilarity measure is an informational divergence. This arbitrary choice is driven by our applications, and is not restrictive. Many other measures could be envisaged, depending on the nature of the data at hand.

If symmetrized, various information divergences [5] can be adopted as a base dissimilarity. In particular, Kullback-Liebler (KL) information divergence is a natural measure of the difference between probability distributions and therefore, after sum-to-one normalization, it can be applied to clustering data whose feature vectors are non-negative, *e.g.*, emission or reflectance spectra [19], gene microarray data [22], hyperspectral images [11], or color images [49]. In order to satisfy the symmetry property required for implementing DRPTs, a

symmetrized version of the Kullback-Leibler divergence is considered as a base dissimilarity measure

$$d_{KLS}(v_i, v_j) = \sum_{z=1}^{l} (\tilde{v_{iz}} - \tilde{v}_{jz}) \log \frac{\tilde{v_{iz}}}{\tilde{v_{jz}}}. \quad (3)$$

The symmetrized Kullback-Leibler divergence is only a semi-metric since it does not satisfy the triangle inequality. However, the DRPT dissimilarity measure (previously defined as the last edge added to the dual rooted MST) still remains a metric according to Appendix A1.

4.2. Benchmark data

Several benchmark comparative tests were performed to demonstrate that EAC-DC clustering is competitive with other state-of-the-art clustering methods. For quantitative performance metrics we used measures based on ground truth reference partitions.

Five popular measures, respectively the Accuracy index (percentage of correctly labelled points, according to a reference partition) the Rand Index (R) [40], the Adjusted Rand index (AR) [28], the Jaccard index (J) [29] and the Normalized Mutual Information (NMI) [43] were implemented. This last one resorts to an information theoretic approach. Note that other information based metrics have been proposed from an axiomatic point of view in [36].

Table 1 summarizes the characteristics of real data sets from UCI Machine Learning Repository [2], namely Breast Cancer Wisconsin (BCW), and Wine dat-sets.

Data set	Number of points	Number of features	Number of clusters
BCW	683	9	2
Wine	178	13	3

Table 1: Characteristics of the Data Sets

The number of classes in each set is known and this knowledge was used by all clustering algorithms in the comparative analysis reported below. In particular,

BCW data have two clusters (that are not well separated) while the Wine data has three clusters. The following algorithms are used for comparison:

- Evidence accumulation clustering with average-link algorithms, referred to as EAC - AL [16][17].
- Graph and hypergraph representation: CSPA, HGPA and MCLA[1] [43].
- Cumulative voting: unnormalized reference-based cumulative voting (URCV), reference-based cumulative voting (RCV), adaptive cumulative voting (ACV) [3].
- Median partition: quadrature mutual information (QMI) [47] (with randomly initialized K-means algorithm to generate the partition ensemble $\mathbf{P}$).
- Diffusion maps (Lafon et al) [34] and commute time (Qiu, Hancock) [39].
- Spectral clustering (Ng et al) [37].
- Dual rooted trees (Symmetric DRPT) [24].

Table 2 reports the relative performance of these different clustering methods on the BCW dataset. The top 9 rows are ensemble averaging methods. EAC-DC was implemented with M fixed at 100 and its numerical score outperforms all other methods on all 5 scoring criteria. Although the RCV, ACV and Diffusion maps clustering algorithms perform nearly as well as EAC-DC in Accuracy, and others come close to EAC-DC according to other criteria, this demonstrates that EAC-DC is competitive with these state-of-the-art clustering algorithms.

Table 3 shows comparative results for the Wine data set. Each data point in Wine is a vector having 13 real valued positive components that may be easily

[1] Codes are available at *http://www.strehl.com*.

interpreted as the characteristic "spectrum" of a wine. Therefore we implemented the EAC-DC clustering algorithm with the symmetrized KL divergence (3) as the base dissimilarity measure for growing the dual rooted MSTs, where $v_{jz}^{\sim}$ is the the spectrum of the j wine for the z-th component. Table 3 shows that this clustering algorithm, denoted EAC-DC (KL) in the table, significantly outperforms all other clustering algorithms regardless of the performance criterion.

In terms of computational load we observed that for the Wine data set the EAC-DC method converged very rapidly as a function of M. In particular, fewer than 20 pairs of roots (initializations of dual rooted trees) were needed to attain its top level of performance. This is to be contrasted to our original symmetric DRPT method, which requires computation of dual rooted trees initialized at all pairs of roots ($N = 178$ leading to $15,931$ different root pairs for the Wine data set). As compared to EAC-DC, SDRPT run time was greater by two orders of magnitude for the tested datasets, as it required to consider $\binom{N}{2}$ pairs of vertices. The computations load of our approach is of same order as other methods reported herein; see [21] for more details.

We also emphasize at this point that a major difference between the proposed approach and k-means and/or k-medoids is the following: there is no need to select a value for k. Here, all partitions are obtained for three classes, that is, 2 rooted clusters + 1 'rejection' (unclassified). This is made possible because all clusters may be non convex, contrary to both k-means and/or k-medoids. This avoids the requirement to set a large k for e.g. k-medoids in the case of strongly non convex clusters. In the next section, clusters may precisely be non convex.

4.3. Astrophysical data

Here, we demonstrate the applicability of the new clustering method to a real world problem in astrophysics. Specifically, we apply the EAC-DC method

to post-AGB (Post Asymptotic Giant Branch) star classification based on data containing information about the infrared (IR) region of the Spectral Energy Distribution (SED). The post-AGB stage is a rather short period in the stellar evolution between the Asymptotic Giant Branch occurring after the end of the hydrogen burning period, and the planetary nebula formation. It is during this period that the spherical symmetry of the circum-stellar environment can be broken and the material can be ionized leading to an asymmetrical planetary nebula. A classification of the different post-AGB can bring crucial information to model the still poorly understood passage from symmetric to asymmetric stages in the last evolution times of intermediate mass stars. The usually admitted reference classification has been provided by van der Veen et al. [13] via a fully supervised approach, which requires hand tuning of the SED clusters, and is therefore based upon experimenter's expertise and prior knowledge.

We obtained a sample from the Toruń catalogue of post-AGB stars ([45]). We used the 344 objects classified as "Very likely post-AGB stars" as of January 2010. We used only the IR region of the data, to avoid bias due to choosing only stars with a counterpart in the visible range. We used data from the 2MASS survey (J, H and K bands), the MSX satellite (8.28 μm, 12.13 μm, 14.65 μm, and 21.3 μm) and the IRAS satellite (12 μm, 25 μm and 60 μm). From the set of 344 spectra, some values were missing at certain wavelengths. We therefore applied linear interpolation, when feasible, leading to a set of 237 complete spectra. Here we focus on clustering the shapes of these spectra, defined as the distribution obtained by normalizing each spectrum with its total energy (sum of each spectrum over all wavelengths). As a base dissimilarity measure we used the symmetrized form of the Kullback-Leibler divergence (3), where $\{\tilde{v}_{jz}\}$ is the j-th normalized spectrum at each wavelength z.

Method EAC-DC(KL) was able to classify the set of 237 post-AGB stars in

9 distinct groups according to their infrared excess in SEDs. The number of clusters is estimated by thresholding the Prim trajectory of the full MST, as described in details in [20]. Figure 7 displays the nine detected clusters, and the resulting homogeneity of the SEDs in each of them seems evident. However, a classification by eye is difficult as several clusters look similar. This justifies to resort to appropriate dissimilarity measures, e.g. KL. For instance, clusters 1 and 8 (if ordered from left to right and top to down) essentially differ in their first spectral lines (threshold effect). Although an astrophysical analysis of each cluster is beyond the scope of this paper, we can qualitatively interpret the results as follows. Each of the nine clusters shows evident homogeneity with a negligible number of interlopers. The clusters exhibit different shape characteristics which are due to different spectral and hence physical properties. From spectral perspective, cluster 6 (black spectra) could be considered as a cluster of rejected spectra. A deeper inspection of each element of this cluster would probably result in further subdivision into subclusters.

Figure (8-a) shows the nine clusters projected in the plane of its two principal components. Note the large degree of overlap in this PCA plane. Figure (8-b) represents a 2 dimensional embedding of the data (each cluster is associated with a symbol and a color), using multidimensional scaling (MDS) [7, 48] applied to the DRPT based distance matrix. Clusters appear more separated in the DRPT+MDS dimensionality reduction as compared to the PCA reduction.

5. Conclusion

In this paper a novel distance measure was introduced, based on hitting times of dual rooted minimal spanning trees, and can be used in any distance based clustering algorithm. The implementation of this new measure has been illustrated in the context of spectral clustering and consensus clustering. The measure was defined as the weight of the longest edge in a graph constructed

by using Prim's algorithm to grow a pair of MSTs, each rooted at a data point, until they contain a common vertex. This dissimilarity measure has been shown to be a metric, regardless of the base dissimilarity measure used for defining edge length in Prim's algorithm. By applying consensus clustering to the dual rooted MSTs grown from a small subset of roots, the computational complexity was significantly reduced. Furthermore, the dual rooted MSTs and consensus clustering combines the advantages of dual rooted trees for discriminating non convex shaped clusters and robustness of consensus based approaches. This led to an algorithm capable of dealing with badly separated, complex shaped clusters, while controlling the computational load.

Due to the universal metric properties of the longest edge dissimilarity measure, the proposed method can be applied in situations where the natural base dissimilarity measure is not a metric, *e.g.* clustering non-Euclidean distributional or spectral types of data. To illustrate this unique feature of our method, we applied it to clustering star spectra using symmetrized KL information divergence. Furthermore, performance comparisons were presented for curated benchmark data, which showed the proposed method to be competitive with other state-of-the-art clustering methods.

As a perspective, it would be worthwhile to explore the use of the proposed metric in other types of distance based clustering algorithms, such as diffusion eigenmaps, multiple linkage clustering, or graph cuts.

6. Acknowledgment

The authors wish to thank Steven Grikschat and Jose Costa for their contributions during the formative stages of this research. This research was partially supported by PPF-ISSO Nice Sophia Antipolis University, France and by the US National Science Foundation under grants CCR-0325571 and CCF-0830490. The authors thank Prof. P. Bendjoya (OCA, University of Nice) for useful

discussions on the application to astrophysical data.

Appendix A1

It is assumed that the base dissimilarity measure is symmetric and satisfies $w(v_i, v_i) = 0, \forall v_i \in V$. To calculate the distance between x and y, $x, y \in V$, we build a DRPT and store the weight of the final constructed edge. The DRPT is obtained by the union of two sub-trees grown from x and y and stopped when they collide (which means that they share one (and only one) vertex). Let w_{final} be the weight of the last constructed edge of the DRPT. The distance is defined as $\delta(x, y) = w_{final}$.

Proposition 1. *$\delta(x, y) = w_{final}$ is a metric.*

Proof: We need show that the following properties are satisfied

- Property 1 : $\delta(x, y) \geq 0$
- Property 2 : $\delta(x, y) = \delta(y, x)$
- Property 3 : $\delta(x, x) = 0$
- Property 4 : $\delta(x, y) \leq \delta(x, z) + \delta(z, y)$,
  $\forall (x, y, z) \in V$

Properties 1, 2 and 3 are straightforward.

The proof of property 4 above is addressed via three lemmas. First we define necessary notation. Denote $V_{x \to y}$ the set of vertices connected in the subtree grown from x and stopped when it collides with the subtree grown from y; let $E_{x \to y}$ denote the corresponding edges. Let $V_{y \to x}$ and $E_{y \to x}$ be defined similarly. Let $T_{x \to y} = \{V_{x \to y}, E_{x \to y}\}$ and $T_{y \to x} = \{V_{y \to x}, E_{y \to x}\}$.

Let h_{xy} be the last connected vertex in the tree rooted at both x and y. Let D be the total number of steps required for $T_{x \to y}$ and $T_{y \to x}$ to collide. Without loss of generality, it will be assumed that h_{xy} is a vertex from $V_{y \to x}$; conversely

its parent vertex verifies $\pi(h_{xy}) \in V_{x\to y}$. Let $w_{final} = w(e(\pi(h_{xy}), h_{xy}))$ be the weight of the last constructed edge. We have the following lemma:

Lemma 1. *Let $\alpha = \max\{w(e) : e \in E_{x\to y} \cup E_{y\to x}\}$, then $w_{final} = \alpha$.*

Proof. Suppose that $\exists e_k \in E_{x\to y} \cup E_{y\to x}$ such that $w(e_k) > \alpha$. Let k be the iteration index at which e_k was created, $V_{x\to y}^{k-1}$ and $V_{y\to x}^{k-1}$ denote the set of vertices connected to x and y respectively after $k-1$ iterations of the DRPT construction. Suppose $k < D$; before $T_{x\to y}$ and $T_{y\to x}$ collide, other edges will be connected to either $V_{x\to y}^{k-1}$ or $V_{y\to x}^{k-1}$, with weight lower than e_k. As, for each step, the constructed edge must be of minimal weight, we conclude that creating the edge e_k violates the construction rules : some lower edges could have been constructed instead. We can thus conclude that α is the weight of the last constructed edge, at iteration D. Note that for $k = 1$, $V_{x\to y}^{k-1}$ and $V_{y\to x}^{k-1}$ are restricted to singletons $\{x\}$ and $\{y\}$ respectively, and the proof still holds. □

By a similar approach, the following lemma can be established :

Lemma 2. *If $v \in V_{x\to y}^{k-1}$ and $\alpha = \delta(x, y)$, then $\delta(v, y) = \alpha$.*

Lemma 3. *Suppose a $D+1$-th step is performed, leading to connect a parent $v \in V_{x\to y} \cup V_{y\to x}$ to a new vertex by an edge of weight β. Then from the construction rule of the DRPT, $\beta > \alpha$.*

Now we split the proof of the triangular inequality into two different cases

- $z \notin V_{x\to y} \cup V_{y\to x}$: z doesn't belong to the path between x and y. $\delta(x, z) \geq \alpha$ (and $\delta(z, y) \geq \alpha$). By Lemmas 1 and 3, $\delta(x, z) > \alpha$ and $\delta(y, z) > \alpha$. Then $\delta(x, y) < 2\alpha < \delta(x, z) + \delta(y, z)$.
- $z \in V_{x\to y}$: by Lemma 2, $\delta(z, y) = \alpha$. As $\delta(x, z) \geq 0$ and $\delta(x, y) = \alpha$, the triangular inequality follows. The same results hold by similar arguments if $z \in V_{y\to x}$. Note that the case where $z = h_{xy}$ can be addressed in a similar manner.

As the four properties are satisfied by δ, then δ is actually a metric. This completes the proof.

Appendix A2

Here we give a proof of property $P4$ from subsection 2.2. Recall that $\mathcal{T}_1$ and $\mathcal{T}_2$ are the rooted MSTs rooted at v_1 and v_2, stopped when they hit each other, in the dual rooted Prim algorithm. Property $P1$ insures that any Prim's algorithm rooted at a vertex from $\mathcal{T}_1 \cup \mathcal{T}_2$ will connect all vertices of $\mathcal{T}_1 \cup \mathcal{T}_2$ before connecting a vertex outside $\mathcal{T}_1 \cup \mathcal{T}_2$. Then, by using property $P2$, it can be concluded that

$$\forall v_i \in \mathcal{T}_1, \forall v_j \in \mathcal{T}_2, \delta(v_i, v_j) = \delta(v_1, v_2)$$

and

$$\forall (v_i, v_j) \in [\mathcal{T}_1 \times \mathcal{T}_1] \cup [\mathcal{T}_2 \times \mathcal{T}_2], \delta(v_i, v_j) \leq \delta(v_1, v_2)$$

Appendix A3

Here we prove the transitivity property of $\mathcal{R}_{v_1}^{v_2}$ that appears in Property $P5$ from Subsection 2.2. For any $x, y \in V$, $x \mathcal{R}_{v_1}^{v_2} y$ means that $\delta(x, y) \leq \delta(v_1, v_2)$. Using property $P4a$ and $P4b$, and notations from Appendix A1 (Lemma 1), we get that

$$\max \{w(e) : e \in E_{x \to y} \cup E_{y \to x}\} \leq \delta(v_1, v_2)$$

Similarly, for any $z \in V$, if $y \mathcal{R}_{v_1}^{v_2} z$, then

$$\max \{w(e) : e \in E_{y \to z} \cup E_{z \to y}\} \leq \delta(v_1, v_2)$$

We now consider a DRPT rooted at x and z. The two inequalities above ensure that there exist a path from x to z that goes through y and uses edges whose measure is always less than $\delta(v_1, v_2)$. Assuming $\delta(x, z) > \delta(v_1, v_2)$ implies that

for some step of the DRPT algorithm described in Section 2.3, an edge of measure larger than $\delta(v_1, v_2)$ is connected, whereas we know that there exists a path from x to z using shorter edges. This is in contradiction to the minimal cost property satisfied by the DRPT. We can conclude that $\delta(x, z) \leq \delta(v_1, v_2)$, and transitivity of the operator $\mathcal{R}_{v_1}^{v_2}$ follows.

Appendix A4. Basics of spectral clustering

Although many flavors of spectral clustering have been proposed, they all share the same algorithmic structure:

1. For a given affinity matrix $A = [a_{ij}]$, define the diagonal matrix $D = diag(A)$ and the graph Laplacian as $L = D - A$.
2. Solve the generalized eigen-value problem

$$Ly = \lambda Dy$$

3. Use the eigenvectors associated with the K smallest positive eigenvalues to determine a K-way partitioning of the data. This can be accomplished by applying K-means to the resulting eigenvectors [37].

The kernel width parameter σ gives the rate at which the affinity between two points decays. While there are many heuristic proposals for selecting the kernel parameter σ, there has been little effort to devise a systematic method for its determination. Complicating this matter, the direct reliance of spectral methods on the affinity matrix can cause clustering results to show high sensitivity to the choice of σ. This may lead to trial-and-error or other heuristic methods involving many re-starts for the selection of σ.

References

[1] T. Asano, B. Bhattacharya, J. Keil, F. Yao, Clustering algorithms based on minimum and maximum spanning trees, in: In Proceedings of the 4th An-

nual ACM Symposium on Computational Geometry, Urbana-Champaign, Illinois, United States, 1988, pp. 252–257.

[2] A. Asuncion, D. J. Newman, UCI machine learning repository (2007). URL http://www.ics.uci.edu/∼mlearn/MLRepository.html

[3] H. G. Ayad, M. S. Kamel, Cumulative voting consensus method for partitions with a variable number of clusters, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (1) (2008) 160–173.

[4] F. R. Bach, M. I. Jordan, Learning spectral clustering, in: Advances in Neural Information Processing Systems, Volume = 16, Editor = S. Thrun and L. Saul and B. Schölkopf, Publisher = MIT Press, Address = Cambridge, MA, Year = 2004, Keyword = BachJordan04:nips spectral clustering.

[5] M. Basseville, Distance measures for signal processing and pattern recognition, Signal Processing 18 (4) (1989) 349–369.

[6] P. Berkhin, Survey of clustering data mining techniques, Tech. rep., Accrue Software (2002).

[7] I. Borg, P. Groenen, Modern Multidimensional Scaling: theory and applications, second edition ed., Springer-Verlag, 2005.

[8] B. Boutsinasa, T. Papastergiou, On clustering tree structured data with categorical nature, Pattern Recognition 41 (12) (2008) 3613–3623.

[9] P. S. Bradley, U. M. Fayyad, Refining initial points for k-means clustering, in: Proceedings of the 15th International Conference on Machine Learning, 1998, pp. 91–99.

[10] L. Breiman, Bagging predictors, Machine Learning Journal 26 (2) (1996) 123–140.

[11] C.-I. Chang, An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis, IEEE Transactions on information theory 46 (5) (2000) 1927–1932.

[12] R. Dejordy, S. Borgatti, C. Roussin, D. Halgin, Vizualizing proxinity data, Field Methods 19 (239).

[13] W. V. der Veen, H. Habing, T. Geballe, Objects in transition from the agb to the planetary nebula stage - new visual and infrared observations, Astronomy and Astrophysics 226 (1989) 108–136.

[14] J. Ding, R.Ma, J.Yang, S.Chen, Tree structured framework for purifying complex clusters with structural roles of individual data, Pattern Recognition 43 (11) (2010) 3753–3767.

[15] S. Dudoit, J. Fridlyand, Bagging to improve the accuracy of a clustering procedure, Bioinformatics 19 (9) (2003) 1090–1099.

[16] A. L. N. Fred, A. K. Jain, Combining multiple clusterings using evidence accumulation, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (6) (2005) 835–850.

[17] A. L. N. Fred, A. K. Jain, Learning pairwise similarity for data clustering, in: Proceedings of International Conference on Pattern Recognition, vol. 1, 2006, pp. 925–928.

[18] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line and an application to boosting, Journal of Computer and System Sciences 1 (1) (1995) 119–139.

[19] L. Galluccio, O. Michel, P. Comon, Unsupervised clustering on multicomponent datasets: Applications on images and astrophysics data, in:

Proceedings of European Signal Processing Conference, Elsevier, Lausanne, Suisse, 2008.

[20] L. Galluccio, O. Michel, P. Comon, A. O. Hero, Graph based k-means clustering, Signal ProcessingPublished online on Jan. 20, 2012.

[21] L. Galluccio, O. Michel, P. Comon, M. Kliger, P. Bendjoya, A. O. Hero, Dual rooted trees for clustering, Tech. rep., GIPSA-Lab/DIS, Grenoble University, France, GIPSA-Lab/DIS, Grenoble University, France. (November 2010).

[22] R. Gentleman, B. Ding, S. Dudoit, J. Ibrahim, Distance measures in DNA microarray data analysis, Bioinformatics and Computational Biology Solutions Using R and Bioconductor (2005) 189–208.

[23] J. Gosh, Multiclassifier systems: Back to the future, in: F. Roli, J. Kitter (eds.), Multiple Classifier Systems, vol. 2364, Springer, 2002, pp. 1–15.

[24] S. Grikschat, J. A. Costa, A. O. Hero, O. Michel, Dual rooted-diffusions for clustering and classification on manifolds, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, Toulouse, France, 2006.

[25] M. Halkidi, Y. Batistakis, M. Vazirgiannis, On clustering validation techniques, Journal of Intelligent Information Systems 17 (2001) 107–145.

[26] T. Hastie, R. Tibshirani, J. Friedman, The Elements of statistical learning, Springer, 2009.

[27] A. Hero, B. Ma, O. Michel, J. Gorman, Applications of entropic spanning graphs, IEEE Signal Processing Magazine 19 (5) (2002) 85–95.

[28] L. Hubert, P. Arabie, Comparing partitions, Journal of Classification 2 (1985) 193–218.

[29] P. Jaccard, The distribution of flora in the alpine zone, New Phytologist 11 (1912) 37–50.

[30] A. K. Jain, R. C. Dubes, Algorithms for Clustering Data, Prentice-Hall, 1988.

[31] A. K. Jain, M. N. Murty, P. J. Flynn, Data clustering: a review, ACM Computing Surveys 31 (3) (1999) 264–323.

[32] L. Kaufman, P. J. Rousseeuw, Clustering by means of medoids, in: Statistical Data Analysis based on the L1 Norm, Elsevier, 1987, pp. 405–416.

[33] L. I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Wiley-Interscience, 2004.

[34] S. Lafon, Y. Keller, R. R. Coifman, Data fusion and multicue data matching by diffusion maps, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1784–1797.

[35] U. V. Luxburg, A tutorial on spectral clustering, Tech. Rep. TR-149, Max-Planck-Institut fur biologische Kybernetik (Aug 2006).

[36] M. Meila, Comparing clustering: an axiomatic view, in: In Proceedings of the 22th International Conference on Machine Learning, Bonn, Germany, 2005, pp. 577–584.

[37] A. Y. Ng, M. I. Jordan, Y. Weiss, On spectral clustering: Analysis and an algorithm, in: Advances on Neural Information Processing Systems, vol. 14, MIT Press, 2001, pp. 849–856.

[38] R. Prim, Shortest connection networks and some generalizations, Bell System Technical Journal 36 (1957) 1389–1401.

[39] H. Qiu, E. Hancock, Clustering and embedding using commute times, IEEE Transactions on Pattern Analysis and Machine Intelligence (2007) 1873–1890.

[40] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical Association 66 (336) (1971) 846–850.

[41] A. Robes-Kelly, E. Hancock, A probabilitic spectral framework for grouping and segmentation, Pattern Recognition 37 (2004) 1387–1405.

[42] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.

[43] A. Strehl, J. Ghosh, Cluster ensemble - a knowledge reuse framework for combining multiple partitions, Journal of Machine Learning Research 3 (2002) 583–617.

[44] W. Stuetzle, A generalized single linkage method for estimating the cluster tree of a density, Journal of Computational and Graphical Statistics 19 (2) (2010) 397–418.

[45] R. Szczerba, N. Siódmiak, G. Stasińska, J. Borkowski, An evolutionary catalogue of galactic post-agb and related objects, Astronomy and Astrophysics 469 (2007) 799–806.

[46] S. Theodoridis, K. Koutroumbas, Pattern Recognition, 3rd ed., Academic Press, 2006.

[47] A. Topchy, A. K. Jain, W. Punch, Clustering ensembles: Models of consensus and weak partitions, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (12) (2005) 1866–1881.

[48] W. Torgerson, Theory and Methods of scaling, Wiley, 1958.

[49] N. Vasconcelos, On the efficient evaluation of probabilistic similarity functions for image retrieval, Information Theory, IEEE Transactions on 50 (7) (2004) 1482–1496.

[50] Z. Wu, R. Leahy, An optimal graph theoretic approach to data clustering: theory and its application to image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (1993) 1101–1113.

[51] B. Xiao, E. R. Hancock, R. C. Wilson, Graph characteristics from the heat kernel trace, Pattern Recognition 42 (11) (2009) 2589–2606.

[52] R. Xu, D. W. II, Survey of clustering algorithms, IEEE Transactions on Neural Networks 16 (3) (2005) 645–678.

[53] H. W. Z. Yu, H. S. Wong, Graph based consensus clustering for class discovery from gene expression data, Bioinformatics 23 (21) (2007) 2888–2896.

[54] C. T. Zahn, Graph-theoretical methods for detecting and describing gestalt clusters, IEEE Transactions on Computers 20 (1) (1971) 68–86.

[55] C. Zhong, D. Miao, R. Wang, A graph-theoretical clustering method based on two rounds of minimum spanning trees, Pattern Recognition 43 (2010) 752–766.

TABLES

Method	Accuracy	Rand	Adjusted Rand	Jaccard	NMI
EAC-DC	**0.9678**	**0.9376**	**0.8743**	**0.9184**	**0.7889**
EAC - AL	0.9429	0.8922	0.7816	0.8488	0.6827
CSPA	0.8448	0.7374	0.4749	0.6523	0.4809
HGPA	0.6501	0.5444	0	0.4856	0
MCLA	0.9575	0.9186	0.8355	0.8869	0.7363
URCV	0.9590	0.9223	0.8409	0.8907	0.7427
RCV	0.9663	0.9348	0.8685	0.9106	0.7755
ACV	0.9649	0.9321	0.8630	0.8067	0.7684
QMI	0.9356	0.8793	0.7561	0.8366	0.6376
SDRPT	0.9414	0.8896	0.7763	0.8453	0.6772
Spectral Clustering (Ng et al.)	0.9356	0.8793	0.7552	0.8313	0.6561
Diffusion Maps (Lafon et al)	0.9605	0.9240	0.8472	0.9025	0.7737
Commute times (Qiu, Hancock)	0.9531	0.9106	0.8191	0.8743	0.7224

Table 2: Results obtained on the Breast Cancer Wisconsin data set; Best scores are indicated in bold font.

Method	Accuracy	Rand	Adjusted Rand	Jaccard	NMI
EAC-DC	0.7022	0.7055	0.3423	0.4407	0.3639
EAC-DC (KL)	**0.8090**	**0.7844**	**0.5248**	**0.5646**	**0.5820**
EAC - AL	0.6910	0.7254	0.3943	0.4672	0.4424
CSPA	0.7135	0.7282	0.3889	0.4685	0.3929
HGPA	0.7022	0.7240	0.3827	0.4635	0.4272
MCLA	0.7247	0.7265	0.3858	0.4693	0.3987
URCV	0.6742	0.6633	0.2789	0.3983	0.3318
RCV	0.6854	0.6759	0.3027	0.4130	0.3484
ACV	0.6966	0.7077	0.3551	0.4409	0.3745
QMI	0.6011	0.6237	0.1686	0.3273	0.2036
SDRPT	0.7135	0.7128	0.3591	0.4499	0.4199
Spectral Clustering (Ng et al.)	0.6966	0.7096	0.3523	0.4440	0.4346
Diffusion Maps (Lafon et al)	0.6573	0.6597	0.3583	0.4320	0.4562
Commute times (Qiu, Hancock)	0.7022	0.7187	0.3711	0.4568	0.4288

Table 3: Results obtained on the Wine data set

FIGURES AND CAPTIONS

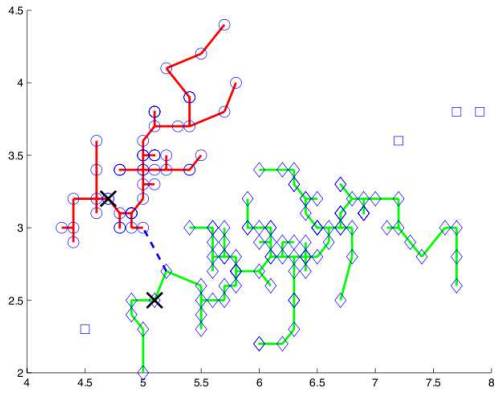


Figure 1: Dual rooted Prim tree built on a data set. Symbol X marks the rooted vertices. The dashed edge is the last connected edge.

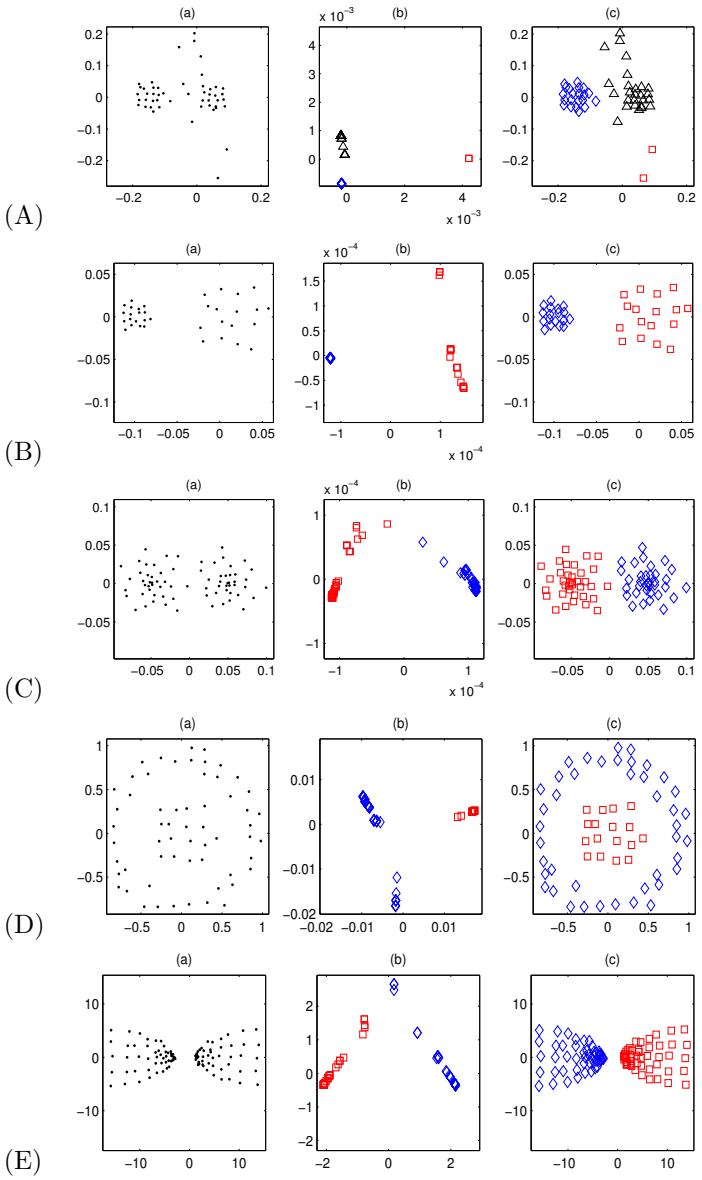


Figure 2: A. 2 identical clusters with outliers; B and E : 2 clusters with different densities; C: 2 identical clusters with variable densities; D: 1 annulus including a cluster. Left(a) : Data; Middle : DRPT-MDS embedded data; right : data labelled with clustering labels obtained by k-means method on the DRPT-MDS embedded data. Note that when outliers are present, k-means was set to search 3 clusters.

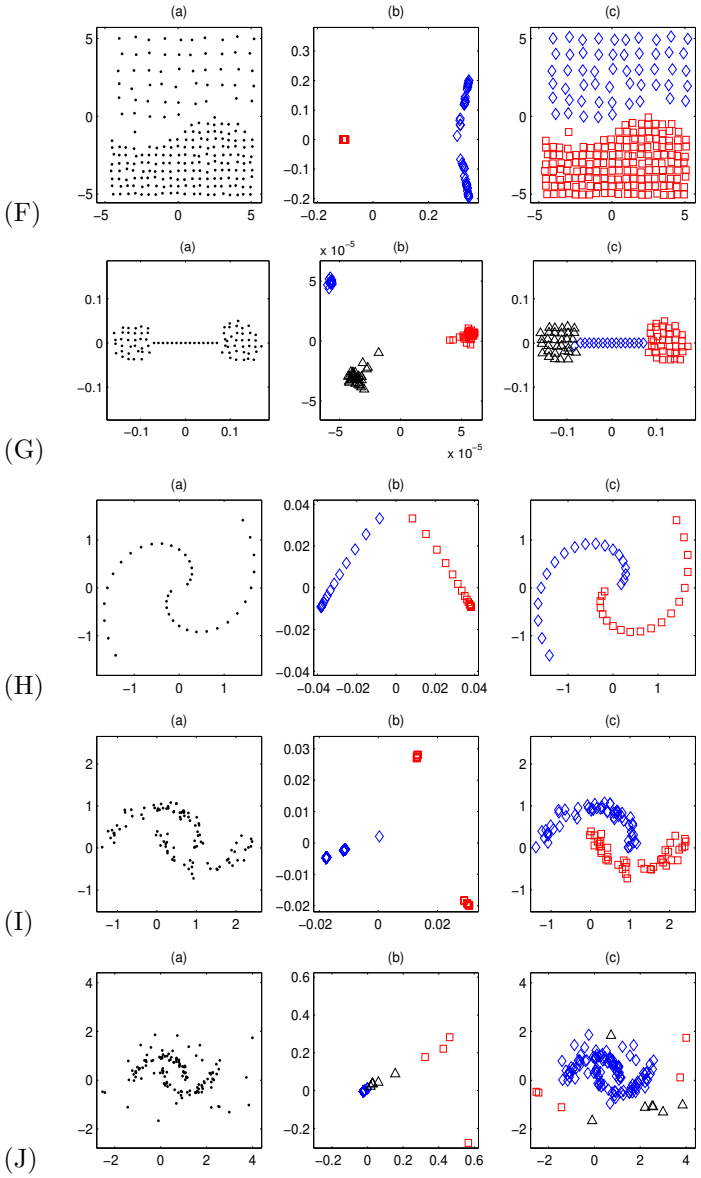


Figure 3: F: Different constant densities; G: 2 clusters and constant density link; H: 2 spirals; I: 2 moons.; J: 2 moons and outliers. Left(a) : Data; Middle : DRPT-MDS embedded data; right : data labelled with clustering labels obtained by k-means method on the DRPT-MDS embedded data. Note that when outliers are present, k-means was set to search 3 clusters.

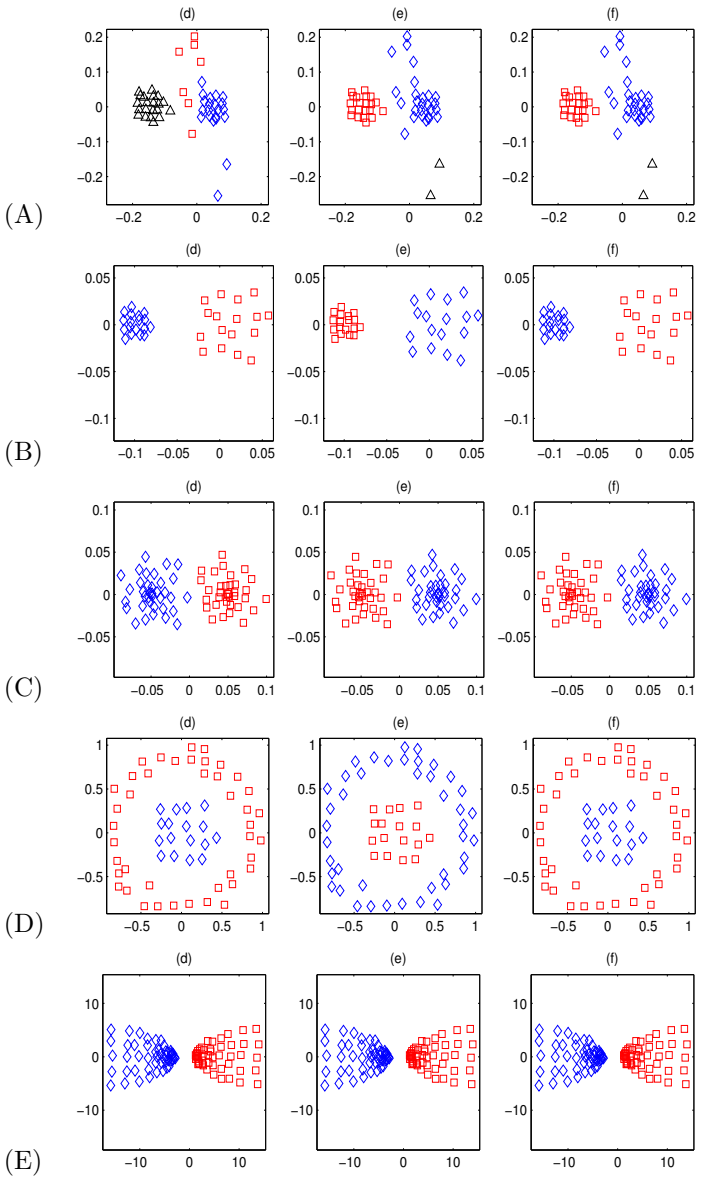


Figure 4: Data sets A to E are those already explored for Figure 2. Left(d): results of spectral clustering with Euclidean metric; Middle(e): Spectral clustering of the complete heat kernelized DRPT consensus matrix; right(f): same as (e), with DRPT consensus matrix estimated from M=N/4 partitions.

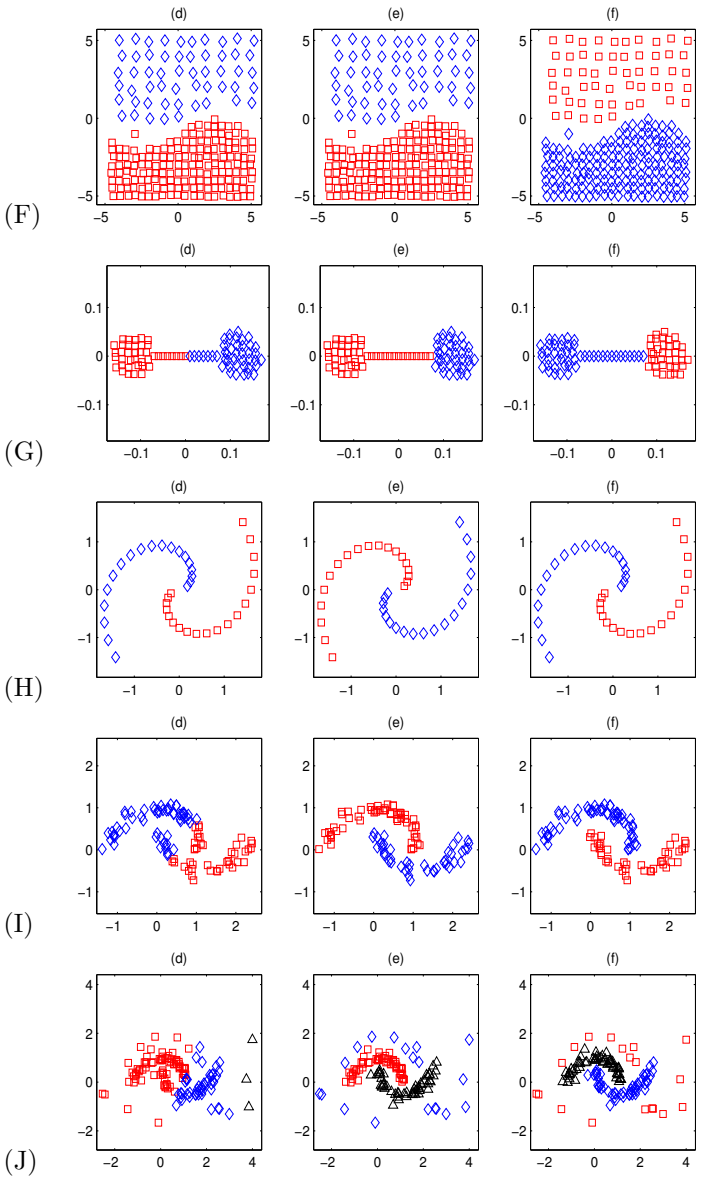


Figure 5: Data sets F to J are those already explored for Figure 3. Left(d): results of spectral clustering with Euclidean metric; Middle(e): Spectral clustering of the complete heat kernelized DRPT affinity matrix; right(f): same as (e), with DRPT affinity matrix estimated from M=N/4 partitions.

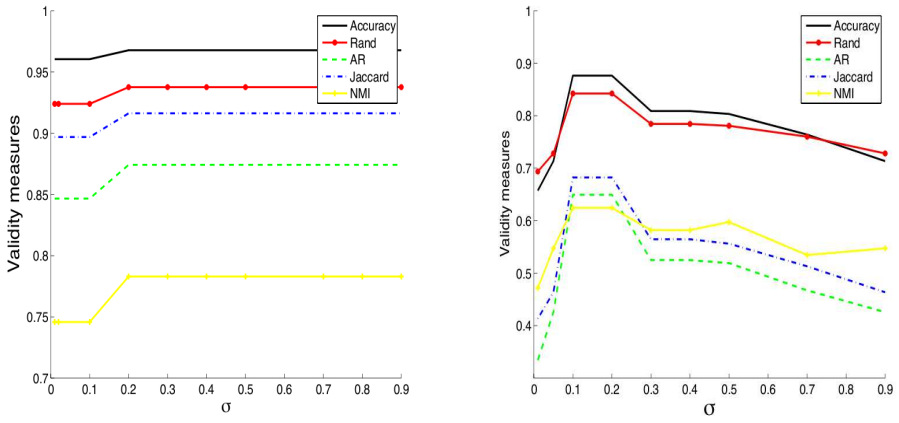


Figure 6: Clustering quality indices as a function of σ/σ_{meas}, for (a) the BCW data and (b) the Wine data. An Euclidean norm is used for constructing the DRPT measure for the BCW data. A KL divergence is used for the DRPT in the case of Wine data set. (σ is expressed as a fraction of σ_{meas}on the plots).

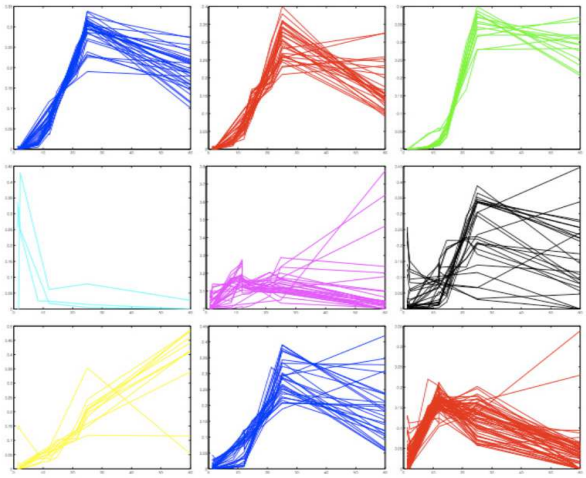
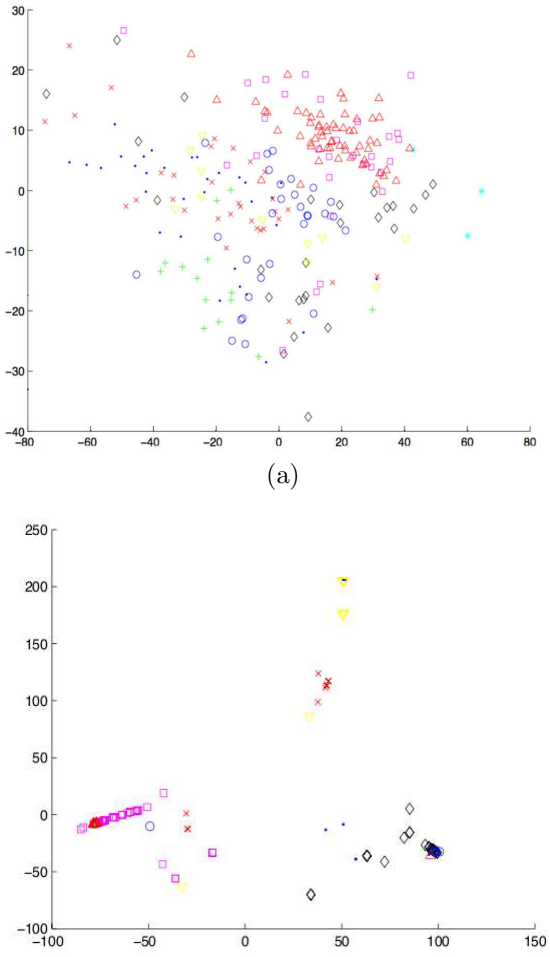


Figure 7: The nine clusters found by method EAC-DC(KL) with a modified *coassoc* criterion. Scales are (from left to right and top to down): 35, 40, 40; 45 80 45; 50 45 35.



(a)

(b)

Figure 8: (a) : The nine clusters found by method EAC-DC(KL) displayed in the plane of the two principal components of higher eigenvalues. (b) : 2D space representation of the data, computed from MDS on the DRPT distance matrix.